UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/679,870	10/06/2003	Patrick Williams	5150-77600	3836

7590 04/30/2007

Jeffrey C. Hood
Meyertons, Hood, Kivlin, Kowert & Goetzel PC
P.O. Box 398
Austin, TX 78767

EXAMINER	
CHEUNG, HUBERT G	

ART UNIT	PAPER NUMBER
2169	

MAIL DATE	DELIVERY MODE
04/30/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No.	Applicant(s)	
	10/679,870	WILLIAMS ET AL.	
	Examiner	Art Unit	
	Hubert Cheung	2169	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *28 February 2007*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-27,29 and 30* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-27, 29 and 30* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☒ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

1. This Office action is in response to the amendment, arguments and remarks, filed on 2/28/2007, in which claims 1-27, 29 and 30 are presented for further examination.

Claims 1-27, 29 and 30 have been amended.

Claim 28 has been cancelled.

Claims 1-27, 29 and 30 are rejected.

Response to Amendment

2. Applicants' arguments filed 2/28/2007 have been fully considered but they are not persuasive. Accordingly, this action has been made FINAL.

Applicants' amendment of the abstract is accepted. The objection to the abstract has been withdrawn.

Applicants' amendment of the paragraph beginning on page 16, line 1 of the specification has been accepted. The objection to the paragraph for minor informalities has been withdrawn.

Applicants' amendments to claims 3 and 14 have been accepted. The objections to claims 3 and 14 for minor informalities have been withdrawn.

Applicants' amendments to claims 1-13, 22, 23, 26, 27 and 29 have been accepted. The rejections of claims 1-13, 22, 23, 26, 27 and 29, under 35 U.S.C. 112, second paragraph, for being indefinite have been withdrawn. The rejections of claims

2-23, 29 and 30, under 35 U.S.C. 112, second paragraph, for being indefinite because they depend on claim 1 have also been withdrawn.

Applicants' amendments to claims 1-25, 29 and 30 have been accepted. The rejections of claims 1-25, 29 and 30, under 35 U.S.C. 101 as being directed to nonstatutory subject matter, have been withdrawn.

Claim Rejections - 35 USC § 102

3. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (e) the invention was described in a patent granted on an application for patent by another filed in the United States before the invention thereof by the applicant for patent, or on an international application by another who has fulfilled the requirements of paragraphs (1), (2), and (4) of section 371(c) of this title before the invention thereof by the applicant for patent.

4. Claims 1-27, 29 and 30 are rejected under 35 U.S.C. 102(e) as being anticipated by Molinari, et al., US 2003/0058280 A1 (hereinafter "Molinari").

For claim 1 (currently amended), Molinari discloses:

A computer-readable memory medium (Molinari, paragraph [0083] where [c]omputer (12) comprises generally the internal components . . . that are typical of personal computers, including besides the CPU a main memory) comprising program instructions executable to:

dynamically determine a plurality of valid parameter values (Molinari, paragraph [0089] when queried by the user . . . the data sink presents to the user a detailed listing of available data sources; and Molinari, paragraph [0148] where [u]pon

the opening of the property page by the user, said property page handler browses front panel Aspect Handler . . . for any data source panels then existing within the front panel);

display a graphical user interface for selecting a parameter value, wherein the graphical user interface visually indicates the plurality of valid parameter values (Molinari, paragraph [0089] where simply by clicking open the property page of the data sink "panel" placed on the desktop by the user[ ], the data sink presents to the user a detailed listing of available data sources; for any said data source selected by the user, said property page then displays a tree view of only those data channels);

receive user input to the graphical user interface to select a first parameter value from the plurality of valid parameter values (Molinari, paragraph [0089] where [u]pon selection of a desired data channel by the user); and

automatically include the first parameter value in source code of a software program in response to the user input selecting the first parameter value (Molinari, paragraph [0037] where [a]s the user places selected virtual instrument "panels" on the desktop and configures their properties . . ., an AIL file containing a description of the selected, created and defined aspects is simultaneously created, including, for each aspect, a description of its properties and connections [i.e., source code]).

For claim 2 (currently amended), Molinari discloses:

The computer-readable memory medium of claim 1, wherein said dynamically determining the plurality of valid parameter values comprises **dynamically determining**

the plurality of valid parameter values based on a configuration of a computer system (Molinari, paragraph [0089] where when queried by the user . . . the data sink presents to the user a detailed listing of available data sources; and Molinari, paragraph [0148] where [u]pon the opening of the property page by the user, said property page handler browses front panel Aspect Handler . . . for any data source panels then existing within the front panel).

For claim 3 (currently amended), Molinari discloses:

The computer-readable memory medium of claim 2, wherein said dynamically determining the plurality of valid parameter values based on the configuration of the computer system comprises **dynamically determining the plurality of valid parameter values based on a hardware configuration of the computer system** (Molinari, paragraph [0089] when queried by the user . . . the data sink presents to the user a detailed listing of available data sources; Molinari, paragraph [0148] where [u]pon the opening of the property page by the user, said property page handler browses front panel Aspect Handler . . . for any data source panels then existing within the front panel; Molinari, paragraph [0180] where the listing of data source types that are, or are to be, connected to the computer [which includes hardware] . . . and that are supported by appropriated device driver software [which inherently means there is hardware]; and Molinari, paragraph [0283] where listed data source types represent all of the various data source types then supported by the software of the invention, and thus usable to connect to a corresponding physical device).

For claim 4 (currently amended), Molinari discloses:

The computer-readable memory medium of claim 3, wherein said dynamically determining the plurality of valid parameter values based on the hardware configuration of the computer system comprises **programmatically examining information regarding the hardware configuration of the computer system** (Molinari, paragraph [0180] where device driver software programmatically examines information regarding hardware; and Molinari, paragraph [0183] where the DAQ Data Source panel [i.e., software] allows a user to set up conventional DAQ hardware device . . . and to configure the subsystems of said DAQ device).

For claim 5 (currently amended), Molinari discloses:

The computer-readable memory medium of claim 3, wherein said dynamically determining the plurality of valid parameter values based on the hardware configuration of the computer system comprises **programmatically querying software associated with one or more hardware devices coupled to the computer system** (Molinari, paragraph [0180] where device driver software programmatically examines information regarding hardware; Molinari, paragraph [0183] where the DAQ Data Source panel [i.e., software] allows a user to set up conventional DAQ hardware device . . . and to configure the subsystems of said DAQ device; and Molinari, paragraph [0283] where listed data source types represent all of the various data source types then supported

by the software of the invention, and thus usable to connect to a corresponding physical device).

For claim 6 (currently amended), Molinari discloses:

The computer-readable memory medium of claim 2, wherein said dynamically determining the plurality of valid parameter values based on the configuration of the computer system comprises **dynamically determining a first plurality of valid parameter values** (Molinari, paragraph [0089] when queried by the user . . . the data sink presents to the user a detailed listing of available data sources; and Molinari, paragraph [0148] where [u]pon the opening of the property page by the user, said property page handler browses front panel Aspect Handler . . . for any data source panels then existing within the front panel);

wherein the program instructions are executable to dynamically determine a second plurality of valid parameter values based on the configuration of the computer system after the configuration of the computer system has been changed (Molinari, paragraph [0266] where once a data source device has been selected, a listing of subsystems associated with said device is presented to the user; and Molinari, paragraph [0291] where [u]pon selecting a particular data source input channel, . . ., the user is provided by the property page of the data source panel with numerous configuration options. . . . The ranges of parameter values presented to the user are set to limits established by the operating specification of the pertinent hardware device).

For claim 7 (currently amended), Molinari discloses:

The computer-readable memory medium of claim 1, wherein said dynamically determining the plurality of valid parameter values comprises **dynamically determining one or more parameter values corresponding to hardware devices coupled to a computer system** (Molinari, paragraph [0291] where [u]pon selecting a particular data source input channel, . . ., the user is provided by the property page of the data source panel with numerous configuration options. . . . The ranges of parameter values presented to the user are set to limits established by the operating specification of the pertinent hardware device);

wherein the first parameter value corresponds to a first hardware device (Molinari, paragraph [0082] where one of the instruments coupled to computer (12), where an instrument is a hardware device; paragraph [0275] where the data acquisition hardware device is a DAQ board; and Molinari, Fig. 2 (16) where a serial instrument is a hardware device);

wherein said automatically including the first parameter value in source code of the software program comprises automatically configuring source code of the software program with a reference to the first hardware device (Molinari, paragraph [0037] where [a]s the user places selected virtual instrument "panels" [i.e., first hardware device] on the desktop and configures their properties . . ., an AIL file containing a description of the selected, created and defined aspects is simultaneously

created, including, for each aspect, a description of its properties and connections [i.e., source code]).

For claim 8 (currently amended), Molinari discloses:

The computer-readable memory medium of claim 1, wherein said dynamically determining the plurality of valid parameter values comprises **dynamically determining one or more parameter values corresponding to resources of one or more hardware devices** (Molinari, paragraph [0291] where [u]pon selecting a particular data source input channel, . . ., the user is provided by the property page of the data source panel with numerous configuration options. . . . The ranges of parameter values presented to the user are set to limits established by the operating specification of the pertinent hardware device);

wherein the first parameter value corresponds to a first resource of a first hardware device (Molinari, paragraph [0275] where the DAQ Controller property page includes a Data Sources drop-down list . . [t]he user's selection of a data source device on this drop-down list effects a connection between said data source device and the DAQ Controller panel);

wherein said automatically including the first parameter value in source code of the software program comprises automatically configuring source code of the software program with a reference to the first resource of the first hardware device (Molinari, paragraph [0037] where [a]s the user places selected virtual instrument "panels" [i.e., first resource of the first hardware device] on the desktop and

configures their properties . . ., an AIL file containing a description of the selected, created and defined aspects is simultaneously created, including, for each aspect, a description of its properties and connections [i.e., source code]).

For claim 9 (currently amended), Molinari discloses:

The computer-readable memory medium of claim 1, wherein said dynamically determining the plurality of valid parameter values comprises **dynamically determining** (Molinari, paragraph [0089] when queried by the user . . . the data sink presents to the user a detailed listing of available data sources; and Molinari, paragraph [0148] where [u]pon the opening of the property page by the user, said property page handler browses front panel Aspect Handler . . . for any data source panels then existing within the front panel) **one or more GPIB resources** (Molinari, paragraph [0082] where Molinari discloses using DAQ resources, which are used in test equipment, through an I/O slot in a computer by using a PCI bus, ISA bus or an EISA bus. Molinari also discloses providing users with a system that is specifically adapted for the development of customized measurement environments for automated test equipment (Molinari, paragraphs [0073]-[0074]). These implications disclose, "one or more GPIB resources.");

wherein the first parameter value comprises a first GPIB resource (Molinari, paragraph [0082] where Molinari discloses using DAQ resources, which are used in test equipment, through an I/O slot in a computer by using a PCI bus, ISA bus or an EISA bus. Molinari also discloses providing users with a system that is specifically adapted

for the development of customized measurement environments for automated test equipment (Molinari, paragraphs [0073]-[0074]). These implications disclose, "wherein the first parameter value comprises a first GPIB resource.");

wherein said automatically including the first parameter value in source code of the software program comprises automatically configuring source code of the software program with a reference to the first GPIB resource (Molinari, paragraph [0037] where [a]s the user places selected virtual instrument "panels" on the desktop and configures their properties . . ., an AIL file containing a description of the selected, created and defined aspects is simultaneously created, including, for each aspect, a description of its properties and connections [i.e., source code]).

For claim 10 (currently amended), Molinari discloses:

The computer-readable memory medium of claim 1, wherein said dynamically determining the plurality of valid parameter values comprises **dynamically determining** (Molinari, paragraph [0089] when queried by the user . . . the data sink presents to the user a detailed listing of available data sources; and Molinari, paragraph [0148] where [u]pon the opening of the property page by the user, said property page handler browses front panel Aspect Handler . . . for any data source panels then existing within the front panel) **one or more Visa resources** (Molinari, paragraph [0082] where Molinari discloses using DAQ resources, which are used in test equipment, through an I/O slot in a computer by using a PCI bus, ISA bus or an EISA bus. Molinari also discloses providing users with a graphical system that is specifically adapted for the

development of customized measurement environments for automated test equipment (Molinari, paragraphs [0073]-[0074]). These implications disclose, "one or more Visa resources.");

wherein the first parameter value comprises a first Visa resource (Molinari, paragraph [0082] where Molinari discloses using DAQ resources, which are used in test equipment, through an I/O slot in a computer by using a PCI bus, ISA bus or an EISA bus. Molinari also discloses providing users with a graphical system that is specifically adapted for the development of customized measurement environments for automated test equipment (Molinari, paragraphs [0073]-[0074]). These implications disclose, "wherein the first parameter value comprises a first Visa resource.");

wherein said automatically including the first parameter value in source code of the software program comprises automatically configuring source code of the software program with a reference to the first Visa resource (Molinari, paragraph [0037] where [a]s the user places selected virtual instrument "panels" on the desktop and configures their properties . . ., an AIL file containing a description of the selected, created and defined aspects is simultaneously created, including, for each aspect, a description of its properties and connections [i.e., source code]).

For claim 11 (currently amended), Molinari discloses:

The computer-readable memory medium of claim 1, wherein said dynamically determining the plurality of valid parameter values comprises **dynamically determining one or more DAQ resources** (Molinari, paragraph [0291] where [u]pon selecting a

particular data source input channel, . . ., the user is provided by the property page of the data source panel with numerous configuration options. . . . The ranges of parameter values presented to the user are set to limits established by the operating specification of the pertinent hardware device [which can include a DAQ resource]);

wherein the first parameter value comprises a first DAQ resource (Molinari, paragraph [0183] where [t]he DAQ Data Source panel allows a user to set up a conventional DAQ hardware device);

wherein said automatically including the first parameter value in source code of the software program comprises automatically configuring source code of the software program with a reference to the first DAQ resource (Molinari, paragraph [0037] where [a]s the user places selected virtual instrument "panels" [i.e., first DAQ resource] on the desktop and configures their properties . . ., an AIL file containing a description of the selected, created and defined aspects is simultaneously created, including, for each aspect, a description of its properties and connections [i.e., source code]).

For claim 12 (currently amended), Molinari discloses:

The computer-readable memory medium of claim 1, wherein said dynamically determining the plurality of valid parameter values comprises **dynamically determining one or more universal resource locators (URLs)** (Molinari, paragraph [0082] where a data source can be obtained from another computer (20) acting as an OPC server. The OPC server can communicate with computer (12) via a wired or wireless means [i.e.,

network]. In order for computer (12) to communicate with computer (20), computer (12) would inherently have to dynamically determine computer (20)'s IP address. A URL is simply a pneumonic for an IP address. Therefore, the invention must inherently be able to dynamically determine one or more IP addresses, which could be URLs);

wherein the first parameter value comprises a first URL (Molinari, paragraph [0082] where a data source can be obtained from another computer (20) acting as an OPC server. The OPC server can communicate with computer (12) via a wired or wireless means [i.e., network]. In order for computer (12) to communicate with computer (20), computer (12) would inherently have to dynamically determine computer (20)'s IP address. A URL is simply a pneumonic for an IP address. Therefore, the invention must inherently be able to have the first parameter value comprise a first URL to reach the data source on computer (20));

wherein said automatically including the first parameter value in source code of the software program comprises automatically configuring source code of the software program with a reference to the first URL (Molinari, paragraph [0037] where [a]s the user places selected virtual instrument "panels" [i.e., first URL] on the desktop and configures their properties . . ., an AIL file containing a description of the selected, created and defined aspects is simultaneously created, including, for each aspect, a description of its properties and connections [i.e., source code]).

For claim 13 (currently amended), Molinari discloses:

The computer-readable memory medium of claim 1, further comprising program instructions executable to:

receive user input specifying filtering criteria for the parameter values (Molinari, paragraph [0149] where [d]ata sources that fit the needs of data sink (130) are checked, and the resulting list of qualifying data sources is displayed, on a combo box, ... Upon the selection of a data source by the user; and Molinari, Fig. 10);

wherein the graphical user interface visually indicates only a subset of the valid parameter values, wherein the subset is determined based on the specified filtering criteria (Molinari, paragraph [0266] where once a data source device has been selected, a listing of subsystems associated with said device is presented to the user; Molinari, paragraph [0291] where [u]pon selecting a particular data source input channel . . . the user is provided by the property page of the data source panel with numerous configuration options. . . . The ranges of parameter values presented to the user are set to limits established by the operating specification of the pertinent hardware device; and Molinari, Fig. 10).

For claim 14 (currently amended), Molinari discloses:

The computer-readable memory medium of claim 1, further comprising program instructions executable to:

receive user input requesting to display the graphical user interface for selecting the parameter value (Molinari, paragraph [0149] where [a] list of panels having a data source aspect is returned, and an aspect filter (135) of data sink aspect

(130) determines, for each data source found, whether the data source accords with the needs of said data sink aspect . . . . Data sources that fit the needs of data sink (130) are checked, and the resulting list of qualifying data sources is displayed, on a combo box, in a "data sources view" of the data sink property page; and Molinari, paragraph [0266] where once a data source device has been selected, a listing of subsystems associated with said device is presented to the user);

wherein said displaying the graphical user interface is performed in response to the user input requesting to display the graphical user interface (Molinari, paragraph [0266] where once a data source device has been selected, a listing of subsystems associated with said device is presented to the user).

For claim 15 (currently amended), Molinari discloses:

The computer-readable memory medium of claim 1, wherein said automatically including the first parameter value in source code of the software program comprises automatically including the first parameter value in one of:

a function call in source code of the software program (Molinari, paragraph [0038] where the user adds more panels, selects more properties, and effects connections between panels, the AIL file expands into a complete textual description of the user's program [i.e., source code]. At the same time, this AIL file corresponds to an executable set of selections from existing libraries of executable code [which inherently includes a function call in the source code]); or

a method call in source code of the software program (Molinari, paragraph [0038] where the user adds more panels, selects more properties, and effects connections between panels, the AIL file expands into a complete textual description of the user's program [i.e., source code]. At the same time, this AIL file corresponds to an executable set of selections from existing libraries of executable code [which inherently includes a method call in the source code]).

For claim 16 (currently amended), Molinari discloses:

The computer-readable memory medium of claim 1, wherein the software program comprises **a graphical program** (Molinari, paragraph [0170] where the user interface should preferably comprise a toolbox of conventional graphical application tools, including graphical means to perform editing and manipulation functions such as cut, copy, save, delete, undelete and the like, ... Additional conventional tools appropriate to any graphical software development application are also preferably provided, including "load project," ["]save project", "save project as", and a "project information" property page);

wherein said automatically including the first parameter value in source code of the software program comprises automatically including the first parameter value in graphical source code of the graphical program (Molinari, paragraph [0032] where the execution of the developed application may therefore be implemented, with no compilation or interpretation of code, by the execution of a simple

textual script file [i.e., source code] identifying the selected attributes, functions and connectivities determined by the users application).

For claim 17 (currently amended), Molinari discloses:

The computer-readable memory medium of claim 16, wherein said automatically including the first parameter value in graphical source code of the graphical program comprises **automatically configuring a node in the graphical program with the first parameter value** (Molinari, paragraph [0037] where [t]he user in the course of programming a measurement application is guided in the selection and the definition of instrument component panels, each of which represents a software "aspect" [i.e., node] having attributes necessary for the user's intended application. As the user places selected virtual instruments "panels" on the desktop and configures their properties . . ., an AIL file [i.e., source code] containing a description of the selected, created and defined aspects is simultaneously being created).

For claim 18 (currently amended), Molinari discloses:

The computer-readable memory medium of claim 17, wherein said automatically configuring the node in the graphical program with the first parameter value comprises **automatically connecting the first parameter value to the node** (Molinari, paragraph [0037] where [t]he user in the course of programming a measurement application is guided in the selection and the definition of instrument component panels, each of which represents a software "aspect" [i.e., node] having attributes necessary for the

user's intended application. As the user places selected virtual instruments "panels" on the desktop and configures their properties . . ., an AIL file [i.e., source code] containing a description of the selected, created and defined aspects is simultaneously being created [i.e., automatically connecting the first parameter value to the node]).

For claim 19 (currently amended), Molinari discloses:

The computer-readable memory medium of claim 17, wherein said automatically configuring the node in the graphical program with the first parameter value comprises **automatically configuring the node to utilize the first parameter value** (Molinari, paragraph [0037] where [t]he user in the course of programming a measurement application is guided in the selection and the definition of instrument component panels, each of which represents a software "aspect" [i.e., node] having attributes necessary for the user's intended application. As the user places selected virtual instruments "panels" on the desktop and configures their properties . . ., an AIL file [i.e., source code] containing a description of the selected, created and defined aspects is simultaneously being created [i.e., automatically configuring the node to utilize the first parameter value]).

For claim 20 (currently amended), Molinari discloses:

The computer-readable memory medium of claim 1, wherein said displaying the graphical user interface comprises **displaying the graphical user interface in a separate window apart from the software program** (Molinari, paragraph [0025]

where [b]y the selections from menu lists, or the "drag and drop" of selected panels icons presented in a "flying tool windows" [which is apart from the main desktop and which is also a part of the graphical user interface of the software program]).

For claim 21 (currently amended), Molinari discloses:

The computer-readable memory medium of claim 1, wherein said displaying the graphical user interface comprises **displaying the graphical user interface in a portion of a program window for the soil-ware program** (Molinari, Fig. 1; Molinari, Fig. 16; and Molinari, Fig. 17).

For claim 22 (currently amended), Molinari discloses:

The computer-readable memory medium of claim 1, **wherein the graphical user interface displays the plurality of valid parameter values as a list** (Molinari, paragraph [0089] where [w]hen queried by the user (simply by clicking open the property page of the data sink "panel" placed on the desktop by the user), the data sink presents to the user a detailed listing of available data sources; for any said data source selected by the user, said property page then displays a tree view of only those data channels that are supported for display by the data sink panel; Molinari, Fig. 11; Molinari, Fig. 13; and Molinari, Fig. 15);

wherein said receiving user input to the graphical user interface to select the first parameter value comprises receiving user input to the graphical user interface to select the first parameter value from the list (Molinari, paragraph [0089]

where [w]hen queried by the user (simply by clicking open the property page of the data sink "panel" placed on the desktop by the user), the data sink presents to the user a detailed listing of available data sources; for any said data source selected by the user, ... Upon selection of a desired data channel by the user, the data sink aspect contains the functionality required to establish automatically a data link between the data source and the data sink).

For claim 23 (currently amended), Molinari discloses:

The computer-readable memory medium of claim 1, wherein said dynamically determining the plurality of valid parameter values includes **dynamically determining one or more property values** (Molinari, paragraph [0089] where [w]hen queried by the user (simply by clicking open the property page of the data sink "panel" placed on the desktop by the user), the data sink presents to the user a detailed listing of available data sources; for any said data source selected by the user, said property page then displays a tree view of only those data channels that are supported for display by the data sink panel; and Molinari, paragraph [0149] where [a] list of panels having a data source aspect is returned, and an aspect filter (135) of data sink aspect (130) determines, for each data source found [i.e., dynamically determining a property], whether the data source accords with the needs of said data sink aspect . . . . Data sources that fit the needs of data sink (130) are checked, and the resulting list of qualifying data sources is displayed [i.e., dynamically determining another property]);

wherein said receiving user input to the graphical user interface to select the first parameter value comprises receiving user input to the graphical user interface to select a first property value (Molinari, paragraph [0089] where [w]hen queried by the user (simply by clicking open the property page of the data sink "panel" placed on the desktop by the user), the data sink presents to the user a detailed listing of available data sources);

wherein the first property value is automatically included in the software program in response to the user input selecting the first property value (Molinari, paragraph [0037] where [a]s the user places selected virtual instrument "panels" on the desktop and configures their properties . . ., an AIL file containing a description of the selected, created and defined aspects is simultaneously created, including, for each aspect, a description of its properties and connections [i.e., source code]; and Molinari, paragraph [0099] where [t]he actions of the user in selecting instrument component panels, and configuring their properties, serve to define the content and properties of the user's intended application program. These actions by the user also operate to create a textual file, called in this description an Aspect Interaction Language (AIL) file, that contains a description of the selected, created and defined software "aspects", ... As the user adds more panels, selects more properties, and effects connections between panels, the AIL file expands into a complete textual description of the user's program).

For claim 24 (currently amended), Molinari discloses:

A computer-readable memory medium comprising program instructions executable to:

determine a plurality of parameter values based on a hardware configuration of a computer system (Molinari, paragraph [0291] where [t]he ranges of parameter values presented to the user are set to limits established by the operating specifications of the pertinent hardware device);

display a graphical user interface for selecting a parameter value, wherein the graphical user interface visually indicates the plurality of parameter values (Molinari, paragraph [0089] where [w]hen queried by the user (simply by clicking open the property page of the data sink "panel" placed on the desktop by the user), the data sink presents to the user a detailed listing of available data sources [i.e., graphical user interface]);

receive user input to the graphical user interface to select a first parameter value from the plurality of parameter values (Molinari, paragraph [0089] where [w]hen queried by the user (simply by clicking open the property page of the data sink "panel" placed on the desktop by the user), the data sink presents to the user a detailed listing of available data sources); and

automatically include the first parameter value in source code of a software program in response to the user input selecting the first parameter value (Molinari, paragraph [0037] where [a]s the user places selected virtual instrument "panels" on the desktop and configures their properties . . ., an AIL file containing a description of the selected, created and defined aspects is simultaneously created, including, for each

aspect, a description of its properties and connections [i.e., source code]; and Molinari, paragraph [0099] where [t]he actions of the user in selecting instrument component panels, and configuring their properties, serve to define the content and properties of the user's intended application program. These actions by the user also operate to create a textual file, called in this description an Aspect Interaction Language (AIL) file, that contains a description of the selected, created and defined software "aspects", ... As the user adds more panels, selects more properties, and effects connections between panels, the AIL file expands into a complete textual description of the user's program).

For claim 25 (currently amended), Molinari discloses:

A computer-readable memory medium comprising program instructions executable to:

determine a plurality of resources of one or more measurement devices coupled to a computer system (Molinari, paragraph [0082] where [t]he system (10) comprises a computer (12), which is connectable to a plurality of instruments [i.e., measurement devices]); Molinari, paragraph [0089] when queried by the user . . . the data sink presents to the user a detailed listing of available data sources; and Molinari, paragraph [0148] where [u]pon the opening of the property page by the user, said property page handler browses front panel Aspect Handler . . . for any data source panels then existing within the front panel);

display a graphical user interface visually indicating a plurality of parameter values, wherein each parameter value corresponds to one of the resources (Molinari, paragraph [0089] where [w]hen queried by the user (simply by clicking open the property page of the data sink "panel" placed on the desktop by the user), the data sink presents to the user a detailed listing of available data sources [i.e., graphical user interface]);

receive user input to the graphical user interface to select a first parameter value from the plurality of parameter values (Molinari, paragraph [0089] where [w]hen queried by the user (simply by clicking open the property page of the data sink "panel" placed on the desktop by the user), the data sink presents to the user a detailed listing of available data sources); and

automatically include the first parameter value in source code of a software program in response to the user input selecting the first parameter value (Molinari, paragraph [0037] where [a]s the user places selected virtual instrument "panels" on the desktop and configures their properties . . ., an AIL file containing a description of the selected, created and defined aspects is simultaneously created, including, for each aspect, a description of its properties and connections [i.e., source code]; and Molinari, paragraph [0099] where [t]he actions of the user in selecting instrument component panels, and configuring their properties, serve to define the content and properties of the user's intended application program. These actions by the user also operate to create a textual file, called in this description an Aspect Interaction Language (AIL) file, that contains a description of the selected, created and defined software "aspects", ...

As the user adds more panels, selects more properties, and effects connections between panels, the AIL file expands into a complete textual description of the user's program).

For claim 26 (currently amended), Molinari discloses:

A system comprising:

a processor (Molinari, paragraph [0083] where [a]ccordingly computer (12) includes at least one central processing unit, or CPU . . . .);

a memory coupled to the processor, wherein the memory stores program instructions (Molinari, paragraph [0083] where [c]omputer (12) comprises generally the internal components . . . that are typical of personal computers, including besides the CPU a main memory);

wherein the processor is operable to execute the program instructions stored in the memory to:

dynamically determine a plurality of valid parameter values (Molinari, paragraph [0089] when queried by the user . . . the data sink presents to the user a detailed listing of available data sources; and Molinari, paragraph [0148] where [u]pon the opening of the property page by the user, said property page handler browses front panel Aspect Handler . . . for any data source panels then existing within the front panel);

display a graphical user interface for selecting a parameter value, wherein the graphical user interface visually indicates the plurality of valid parameter

values (Molinari, paragraph [0089] where simply by clicking open the property page of the data sink "panel" placed on the desktop by the user[ ], the data sink presents to the user a detailed listing of available data sources; for any said data source selected by the user, said property page then displays a tree view of only those data channels);

receive user input to the graphical user interface to select a first parameter value from the plurality of valid parameter values (Molinari, paragraph [0089] where [w]hen queried by the user (simply by clicking open the property page of the data sink "panel" placed on the desktop by the user), the data sink presents to the user a detailed listing of available data sources); and

automatically include the first parameter value in source code of a software program in response to the user input selecting the first parameter value (Molinari, paragraph [0037] where [a]s the user places selected virtual instrument "panels" on the desktop and configures their properties . . ., an AIL file containing a description of the selected, created and defined aspects is simultaneously created, including, for each aspect, a description of its properties and connections [i.e., source code]; and Molinari, paragraph [0099] where [t]he actions of the user in selecting instrument component panels, and configuring their properties, serve to define the content and properties of the user's intended application program. These actions by the user also operate to create a textual file, called in this description an Aspect Interaction Language (AIL) file, that contains a description of the selected, created and defined software "aspects", ... As the user adds more panels, selects more properties, and effects connections

between panels, the AIL file expands into a complete textual description of the user's program).

For claim 27 (currently amended), Molinari discloses:

A method for modifying source code of a software program, the method comprising:

dynamically determining a plurality of valid parameter values (Molinari, paragraph [0089] when queried by the user . . . the data sink presents to the user a detailed listing of available data sources; and Molinari, paragraph [0148] where [u]pon the opening of the property page by the user, said property page handler browses front panel Aspect Handler . . . for any data source panels then existing within the front panel);

displaying a graphical user interface for selecting a parameter value, wherein the graphical user interface visually indicates the plurality of valid parameter values (Molinari, paragraph [0089] where simply by clicking open the property page of the data sink "panel" placed on the desktop by the user[ ], the data sink presents to the user a detailed listing of available data sources; for any said data source selected by the user, said property page then displays a tree view of only those data channels);

receiving user input to the graphical user interface to select a first parameter value from the plurality of valid parameter values (Molinari, paragraph [0089] where [w]hen queried by the user (simply by clicking open the property page of

the data sink "panel" placed on the desktop by the user), the data sink presents to the user a detailed listing of available data sources); and

automatically including the first parameter value in source code of the software program in response to the user input selecting the first parameter value (Molinari, paragraph [0037] where [a]s the user places selected virtual instrument "panels" on the desktop and configures their properties . . ., an AIL file containing a description of the selected, created and defined aspects is simultaneously created, including, for each aspect, a description of its properties and connections [i.e., source code]; and Molinari, paragraph [0099] where [t]he actions of the user in selecting instrument component panels, and configuring their properties, serve to define the content and properties of the user's intended application program. These actions by the user also operate to create a textual file, called in this description an Aspect Interaction Language (AIL) file, that contains a description of the selected, created and defined software "aspects", ... As the user adds more panels, selects more properties, and effects connections between panels, the AIL file expands into a complete textual description of the user's program).

For claim 29 (currently amended), Molinari discloses:

The computer-readable memory medium of claim 1, **wherein said displaying the graphical user interface that visually indicates the plurality of valid parameter values is performed while a user is creating the software program** (Molinari,

paragraph [0089] [w]hen queried by the user . . . the data sink presents to the user a detailed listing of available data sources);

wherein said automatically including the first parameter value in source code of the software program (Molinari, paragraph [0037] where [a]s the user places selected virtual instrument "panels" on the desktop and configures their properties . . ., an AIL file containing a description of the selected, created and defined aspects is simultaneously created, including, for each aspect, a description of its properties and connections [i.e., source code]) **is performed to aid the user in creating the software program** (Molinari, paragraph [0019] where [u]sing the tools provided by the invention, a user having essentially no software programming experience is enabled to design and construct a working, problem-specific measurement solution).

For claim 30 (currently amended), Molinari discloses:

The computer-readable memory medium of claim 1, wherein the software program comprises a **graphical program** (Molinari, paragraph [0318] where a preferred embodiment of the present invention [has] an internal set of "Windows" classes, including the functionality to create a desktop window that sits "on top" of the original Windows operating system; Molinari, Fig. 1; Molinari, Fig. 11; Molinari, Fig. 13; and Molinari, Figs. 16-25);

wherein said automatically including the first parameter value in source code of the software program (Molinari, paragraph [0037] where [a]s the user places selected virtual instrument "panels" on the desktop and configures their properties . . .,

an AIL file containing a description of the selected, created and defined aspects is simultaneously created, including, for each aspect, a description of its properties and connections [i.e., source code]) **comprises automatically including the first parameter value in a block diagram of the graphical program** (Molinari, Fig. 1 where the blocks are (42), (44) and (46); Molinari, Fig. 17 where the blocks are (212), (214) and (221); and Molinari, Fig. 25).

Response to Arguments

5. Applicants' argument, see page 13, lines 8-13, with respect to the use of the trademarks, SPARC and CVI, have been fully considered and are persuasive. The objection to the use of the trademarks, SPARC and CVI, has been withdrawn.

However, applicants' argument, see page 13, lines 8-13, with respect to the use of the trademarks VISUAL C++, VISUAL BASIC, JAVA, LABVIEW, DASYLAB, DIADEM, MATRIXX, SYSTEMBUILD, SIMULINK, WIT, SANSCRIPT, OBJECTBENCH, HYPERSIGNAL, PENTIUM, POWERPC, LABWINDOWS and DELPHI have been fully considered but they are not persuasive. The objection will be reiterated and finalized in the instant Office Action.

The use of the trademarks VISUAL C++, VISUAL BASIC, JAVA, LABVIEW, DASYLAB, DIADEM, MATRIXX, SYSTEMBUILD, SIMULINK, WIT, SANSCRIPT, OBJECTBENCH, HYPERSIGNAL, PENTIUM, POWERPC, LABWINDOWS and DELPHI has been noted in this application (i.e., page 3, line 10). They should be capitalized wherever they appear and be accompanied by the generic terminology.

> Trademarks should be identified by capitalizing each letter of the mark (in the case of word or letter marks) or otherwise indicating the description of the mark (in the case of marks in the form of a symbol or device or other nontextual form). MPEP 608.01(v).

Although the use of trademarks is permissible in patent applications, the proprietary nature of the marks should be respected and every effort made to prevent their use in any manner which might adversely affect their validity as trademarks.

Applicants argue that Molinari does not disclose to, "automatically include the first parameter value in source code of a software program in response to the user input selecting the first parameter value" (middle of page 14 and the top of page 15 of the instant remarks) as recited in currently amended claim 1. Applicants further argue that Molinari explicitly discloses that the AIL file is not source code (bottom of page 14 of the instant remarks).

Examiner respectfully disagrees.

Source code is defined as, "***[h]uman-readable program statements*** written by a programmer or developer in a high-level or assembly language that are not directly readable by a computer" (n.). *Microsoft® Computer Dictionary, Fifth Edition*. Microsoft Press, 2002.).

The examiner would like to point to Molinari, paragraph [0019], where Molinari discloses an invention that presents the user with a graphical desktop for developing software applications through the use of graphical programming techniques including the manipulation of icons and dropdown lists by using a mouse. "[A] user having essentially no software programming training or experience is enabled to design and

construct a working, problem-specific measurement solution which can readily be saved to a file, and be exported to one or a plurality of end user personal computers." Molinari achieves this by inserting the user's selected objects and parameters, derived from the user's manipulation of icons and dropdown lists through the use of a mouse, into a computer file which is then interpreted by the invention and turned into a working computer program. This is clearly generating a form of source code used to create the computer program. The computer file that is interpreted by the invention is or essentially is source code, as defined above.

Furthermore, Molinari, in paragraph [0037], as cited by the examiner in the previous Office action, discloses "an AIL file containing a ***description of the selected, created and defined aspects*** . . . including for each aspect, a ***description of its properties and connections***." Molinari, in paragraph [0038], further discloses the AIL file as being "***human-readable***, and readily saved and stored." Molinari, in paragraph [0038], also discloses the AIL file as being "a complete ***textual description of the user's program***." Essentially, the AIL file is a human-readable, high-level, textual description of the program. In other words, the AIL file is or essentially is source code, as defined above.

Finally, in the very paragraph applicants cite in their argument, paragraph [0036], Molinari discloses the use of other textual representations for aspects, such as using extensible markup language ("XML"). XML is a subset of the Standard Generalized Markup Language ("SGML"). Hypertext Markup Language ("HTML") is also a subset of SGML. HTML is the ***source code*** that makes up web pages. Therefore, XML, being a

subset of SGML, is also source code for web pages. Molinari discloses the use of XML to insert "aspects" to create computer programs. Therefore, Molinari discloses the insertion of source code to create computer programs.

Applicants' argue that independent claims 24-27 should be allowed for similar reasons to the reasons for the allowance of claim 1 (top of page 15 of the instant remarks).

Examiner respectfully disagrees.

Claims 24-27 are rejected for reasons similar to the rejection of amended claim 1 above.

Conclusion

THIS ACTION IS MADE FINAL. Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Point of Contact

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Hubert Cheung whose telephone number is 571-270-1396. The examiner can normally be reached on M-R 7:30A - 5:00P EST; alt. F 7:30A - 4:00P EST.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Christian Chace can be reached on 571-272-4190. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

Examiner: Hubert Cheung HC 4/24/07
Date: April 17, 2007

CHRISTIAN CHACE
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100